INTERNAL DOCUMENT 120

I.O.S.

SPECTRAL ANALYSIS OF WAVE DATA

A description of the computer programs developed for use in wave climate studies at I.O.S.

NATURAL ENVIRONMENT RESEARCH COUNCIL
INSTITUTE OF OCEANOGRAPHIC SCIENCES

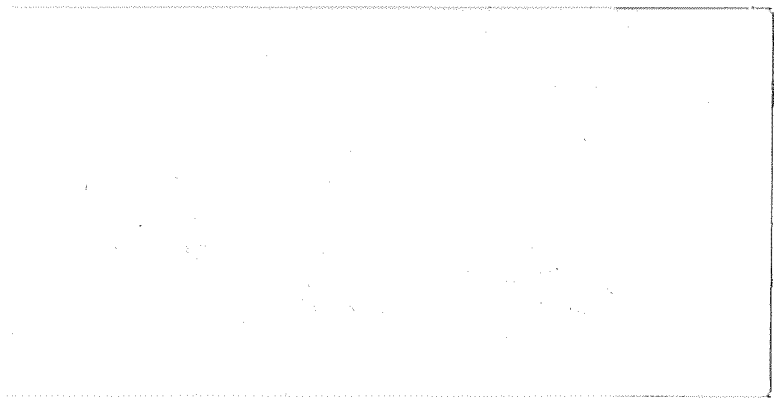
INSTITUTE OF OCEANOGRAPHIC SCIENCES

Wormley, Godalming,
Surrey GU8 5UB
(042-879-4141)

(Director: Dr. A. S. Laughton)

Bidston Observatory,
Birkenhead,
Merseyside L43 7RA
(051-653-8633)
(Assistant Director: Dr. D. E. Cartwright)

Crossway,
Taunton,
Somerset TA1 2DW
(0823-86211)
(Assistant Director: M. J. Tucker)



SPECTRAL ANALYSIS OF WAVE DATA

A description of the computer programs developed for use in wave climate studies at I.O.S.

Internal Document No 120

E G Pitt
Institute of Oceanographic Sciences
Crossway
Taunton
Somerset

April 1981

CONTENTS

IOS ROUTINE WAVE DATA SPECTRAL ANALYSIS PROGRAM

1.0 Introduction

The system of programs to be described in this document is used to calculate the spectra of wave records which are collected as part of IOS's wave climate studies. Accordingly, the main program is written to process wave observations which consist of 2048 digital points, and which are made at 3-hourly intervals, and in this sense is entirely specific. However, the program is modular in structure and many of the parameters of the calculations are either input at each run or can be otherwise easily modified, so that there is considerable flexibility.

2.0 Analysis Scheme

The analysis method is as described in the note by Pitt (1980), attached as appendix A, and is centred around a FastFourier Transform subroutine due to R.C. Singleton of the Stanford Research Institute.

The program performs the following calculations on each observation (see section 4 of appendix A):

(1) Calculates and subtracts the mean of the data
(2) Cosine tapers the data series (optional)
(3) Calculates discreteFourier Transform of the data
(4) Forms the periodogram
(5) Adjusts the scaling for the effect of the cosine taper
(6) Applies frequency response corrections
(7) Calculates the moments
(8) Forms the smoothed spectrum
(9) Calculates the spectral peakedness parameter, Qp.

3.0 Inputs

The input files must consist of a series of digital wave observations in ascending time order, each observation being labelled with its start date and time, and separated by 3 hours from its neighbours. The format of the input files is discussed in the description of SUBROUTINE READ (appendix C).

4.0 Outputs

The program writes an output file containing the spectra for the period specified, and a shortened version of this file for use as an index. Only one

pair of output files is produced at each run. The output files may be specified for one calendar month, which is the practice at IOS, or any shorter or longer period; except that the files always end with the nominal 2100 hours observation on the end-day specified.

The format of the output files is described in appendix F.

5.0 The main program

All the calculations listed above are carried out by subprograms, and so the function of the main program is to organise the flow of data from the input file through the various subprograms to the output file.

Figure 1 shows a simplified flow diagram of the program; the functions of the several blocks will be discussed in the following paragraphs.

(Full instructions on running the program are included at appendix B.)

5.1 Input of run data

At the start of the program information needed to identify the data to be processed in this particular run is input. This includes the nominal* date and time of the start of the period for which the observations are to be processed, the date of the end of the period and the number of input files.

The program operates on the assumption that the wave observations were made at 3-hourly intervals and these would normally (but not necessarily) correspond to the intermediate synoptic hour sequence ie midnight, 0300, 0600, 0900, 1200, 1500, 1800 and 2100 hours GMT of every day. As each midnight is passed, the date (expressed as the Julian day number) is incremented by one and the sequence repeated.

The nominal date and time of the start of the period serves to indicate the start of the sequence of nominal times, and would thus normally (but not necessarily) be one of the intermediate synoptic hours.

5.2 Status override facility

The observations which form the input of the program will normally have been subjected to an automatic validation procedure. In the IOS system the outcome of this procedure is indicated by a series of ten error flags which are contained in array KFLAG, and one of these flags, KFLAG(9), is used as an

---

* It is essential to maintain a sequence of nominal times to which the actual times of the observations are related so that those periods for which there are no observations can be correctly represented in the output file.

indicator of whether or not the observation as a whole is valid. It has been found useful to be able to manually override this status flag on specific observations where further investigation shows that the automatic system has made the wrong decision. This facility allows KFLAG (9) for a limited number of specified observations to be set to 1 or 0 as required.

5.3 Sampling interval

The interval between individual sample points in the observations is specified in seconds as SINT.

5.4 Logical End of File Codes, Scale Factors

It is usual when making series of oceanographic measurements to record any changes to the observational method which could possibly affect the data. One way of indicating such changes is to divide the measurements series at the time each change was made: the Nth series ending before the change and the (N + 1)th series starting after it.

The output files are terminated by a logical end of file code which is written as groups of 9 to indicate the end of a standard data period (eg one calendar month) or as groups of 8 to indicate the end of a data series. As an example of the use of the logical end of file code the IOS practice will be described:

Wave measurements are made at a single geographical position over a period of some years, often using Waverider buoys. A freshly-serviced and calibrated replacement buoy is deployed at regular intervals which are of order 3 to 6 months. Now, the output files each consists of data from exactly one calendar month, and since buoy replacements rarely if ever exactly coincide with the end of the month, breaking the series at each deployment would result in an unacceptably fragmented output series. In order to maintain the integrity of the monthly spectral and index files, the logical end of file code is set to 8 at the beginning of the month containing the redeployment or final recovery or other significant change.

The scale factor required to convert the digital wave measurements to 'user units' (usually meters) is input as SCALF. Since the scaling will in general change at the end of a series, if the logical end of file code is specified as 8 provision is made for the input of a second scale factor, SCALF2, together with the precise date and time from which it is to be applied.

The way the output files are organised and the way the logical end of file codes are used will depend on the nature of the wave measuring project. For example, another scheme might be to produce one pair of output files from one run of the program for each deployment period. The logical end of file code would then always be 8.

5.5 Taper Scale

The program allows the application of a variable taper function to the digital observation before the Fast Fourier Transform is performed. A cosine taper is applied to a fraction $^{1}/_{NPR}$ of the digital observation at each end. Thus NPR = 2 results in a full cosine taper. NPR is usually specified as 8 in the IOS system, resulting in a taper being applied to $\frac{1}{8}$ of the digital observation at each end. If no taper is required, NPR is set to 0.

5.6 Smoothing of the spectral estimates

In order to improve the statistical stability of the final spectrum, the elementary spectral estimates are smoothed by averaging over blocks of adjacent estimates NAV at a time. NAV sets the resolution of the spectrum as well as the stability: with an observation length of 1024 seconds, NAV = 10 has been found to be a reasonable choice.

5.7 Initialisation of look-up tables

A number of arrays are initialised by calls to subroutines early in the program. The arrays and their contents are:-

ISILU - Table of values required to perform cosine taper. If NPR = 0 the array is not initialised.

FLUP - table of frequencies of the unsmoothed spectrum (periodogram).

FAVTAB - table of frequencies for the smoothed (final) spectrum.

CORLUP - table of frequency-dependent corrections to be applied to the periodogram.

The first three are completely automatic, but some discussion of the fourthis required.

CORLUP is initialised by a call to SUBROUTINE FCORR which forms the correction as the square of the product of two amplitude correction functions. These FUNCTION subprograms are input as arguments of FCORR and must be declared in an EXTERNAL statement in the main program.

Typically the correction functions might be that appropriate to the Waverider buoy double integrator and that appropriate to the receiving/recording system. If only one function is required, the remaining argument is set to UNITY, which is a FUNCTION subprogram which always returns the value 1.0.

5.8 Reading from the data file

The input files are ascribed Logical Unit Numbers 10 and upwards and are read by SUBROUTINE READ. The SUBROUTINE supplied is, of course, highly specific to the IOS system, but by studying the detailed description of READ given in Appendix C it should be possible to write a replacement for READ which is compatible with the main program and with the user's validation program output.

As the end of each file is reached, the logical unit number of the input stream is incremented by one and the next file is read.

This process is repeated until all data for the period specified have been processed.

It is sometimes useful to be able to set the end of the output file at a date later than the end of the available data. This is an essential feature when the output files are for a standard period such as one calendar month since it will not infrequently happen that periods of data loss include the end of a month. In such cases, when all of the input files specified have been read, the remaining days in the specified period appear in the output files filled with "no-data" entries. In order to achieve this, the program needs to know how many input files are required to cover the specified period (NIMF).

5.9 Observation time checking

The allocation of each observation to one of a series of nominal times, and the detection of missing records will be discussed.

Study of the flow chart (Figure 1) will give an indication of how the program operates to produce the following effects:-

(1) The first observation encountered whose time falls within $\pm$90 minutes of the nominal time is processed and appears in the output file with its actual time.

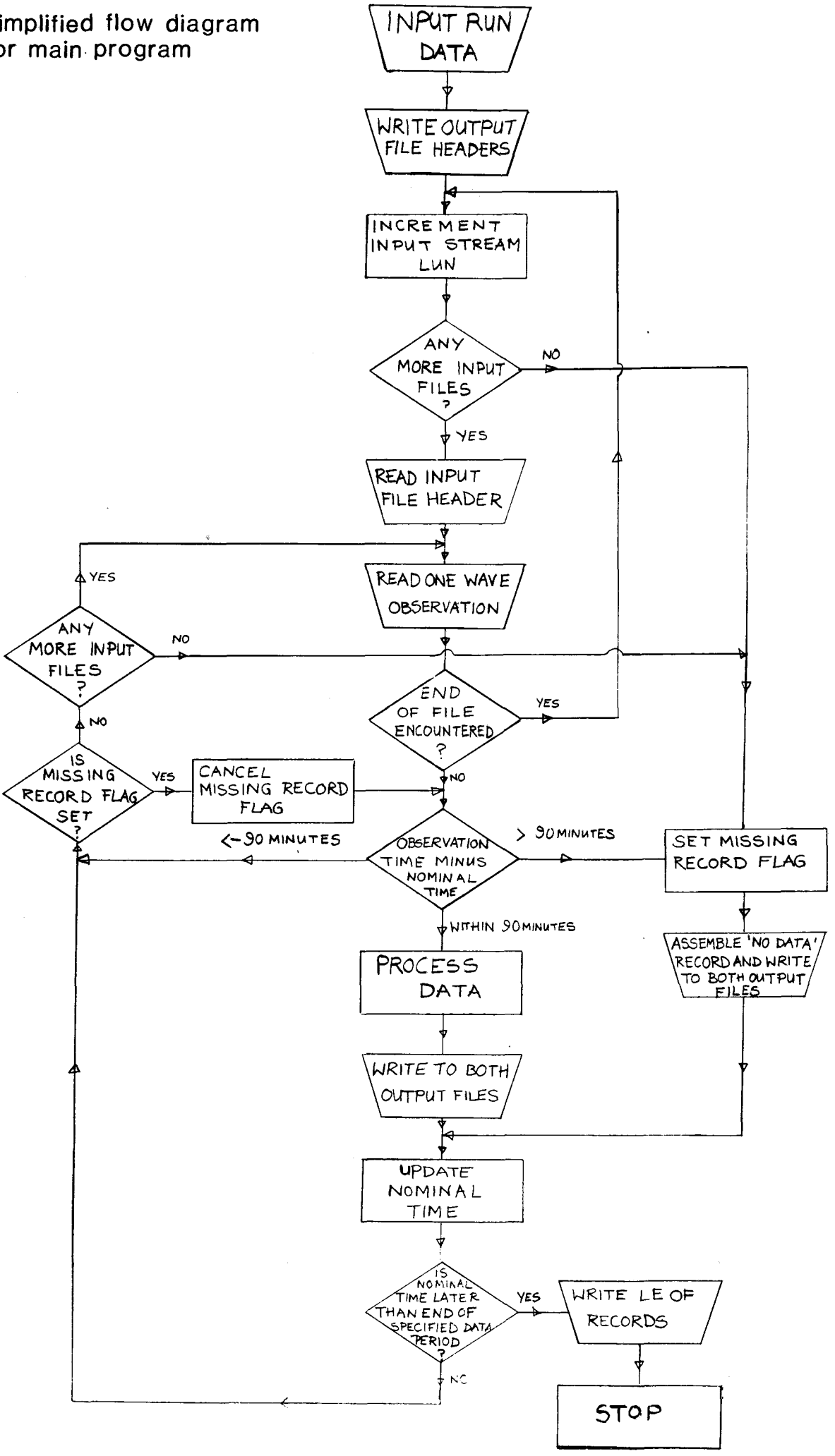
(2) Only one observation is allocated to each nominal time.

(3) If no observation is available for a particular nominal time, a "no-data" entry is made in the output files against that nominal time.

The program expects the day number which appears in the input file to increment correctly at each midnight, and if the data recorder clock is "perpetual" ie it resets the day number to 1 at the beginning of each year, paying due regard to leap years, the program will operate correctly at year changes. However, data loggers rarely have a perpetual clock, so that the program has been designed to operate correctly at year changes so long as the day number of the logger clock is (correctly) reset sometime within the new year. The observation allocation logic operates independently of the year number recorded in the input file.

Figure 1

Simplified flow diagram for main program



NOTES ON

THE SPECTRAL ANALYSIS OF WAVE RECORDS

by

E G PITT

Institute of Oceanographic Sciences
Crossway
Taunton
April 1980

CONTENTS

NOTES ON THE SPECTRAL ANALYSIS OF WAVE RECORDS

List of symbols

a_n n th Fourier amplitude (cosine) of sample record

b_n n th Fourier amplitude (sine) of sample record

C_n Amplitude of the n th component of the frequency domain representation of a random process

$\hat{C}_n$ Amplitude of the n th component of the frequency domain representation of the sample record

ϵ_r Normalised standard error of a spectral estimate

θ_n Phase of the n th component of a random process

f Frequency

f_s Sampling frequency

f_{max} Maximum frequency present in the sample wave record

Δf Difference between frequencies of successive spectral estimates

f_0 Fundamental frequency of Fourier analysis

f_n n th harmonic of the fundamental frequency

f_i The frequency of the i th smoothed spectral estimate above zero frequency

H_s Significant wave height

l (Number of elementary sample spectral estimates included in each smoothed estimate) - frequency domain averaging

k (Number of elementary sample spectral estimates included in each smoothed estimate) - time domain averaging

n Harmonic number in sample Fourier series

N Number of measurements of waveheight in a sample record

Q_p Spectral peakedness parameter

ϕ_n Elementary (sample) spectral estimate corresponding to the n th harmonic of the sample record.

S_i The i th smoothed spectral estimate above zero frequency

S Spectrum of a random process

T Sample record length

ΔT Sampling interval between individual measurements

$\overline{T}_z$ Mean zero-crossing period

$\overline{T}_e$ Energy period

NOTES ON THE SPECTRAL ANALYSIS OF WAVE RECORDS

1. INTRODUCTION

If we measure the height, h, of the sea surface at a single geographical position as a function of time, t, we can model the resulting process as the linear superposition of an infinite number of sinusoids of small amplitude (see Cartwright and Longuet-Higgins, 1956):

$$h(t) = \sum_{n=1}^{\infty} C_n \cos(2\pi f_n t + \theta_n) \qquad (1.1)$$

where the frequencies f_n are densely distributed in the interval $(0, \infty)$ and the phase angles, θ_n, are random and uniformly distributed in the range $(0, 2\pi)$.

The amplitudes C_n are such that in any small interval of frequency df,

$$\sum_{f_n=f}^{f+df} \tfrac{1}{2} C_n^2 = S(f)\,df \qquad (1.2)$$

where $S(f)$ is called the (variance) spectrum of $h(t)$. We see that the spectrum is the frequency-domain description of a linear random process. Note that it is defined as a density and has units of variance per unit bandwidth. Since the variance is proportional to the energy per unit area, S is often called the energy spectrum of h.

The practical utility of the spectrum will not be discussed in this note, except that the definitions of some common wave parameters in terms of spectral quantities will be given.

1.1 The Practical Problem:

The problem that the wave observer must address is to estimate the spectrum from a finite sample of the process, and to state for the benefit of the user how closely this estimate can be expected to approximate the true spectrum of the sea on the occasion of the sample measurement.

The outline which now follows is based on present practices in routine wave data processing in the Applied Wave Research Group in IOS, and while they are believed to be sound they are not necessarily optimum for any given application. Moreover, they are liable to change as improved methods are developed.

2. SAMPLING SCHEME

2.1 The wave height signal should be digitised at a rate which is high enough to describe the highest frequency component present. According to the sampling theorem, if the highest frequency component in the data is f_{max}, the sampling rate, f_s, is given by

$$f_s = \frac{1}{\Delta T} \geq 2 f_{max} \qquad (2.1)$$

Often, f_{max} is defined by including a lowpass (anti-aliasing) filter in the signal path prior to digitisation.

2.2 The wave height record should be long enough to allow the spectrum to be estimated with adequate statistical reliability. The standard error of the spectral estimate, ϵ_r, is given by

$$\epsilon_r = \sqrt{1/T\Delta f} \qquad (2.2)$$

where T is the length of the record, and Δf is the frequency separation of successive spectral estimates (the 'resolution' of the spectrum). For fixed Δf the error is proportional to $1/\sqrt{T}$, so that we should record as long a record as possible. A theoretical limitation is the need for the wave record to be statistically stationary over the sample period.

Practical limitations concern the available recording capacity and power. A fuller discussion of sampling errors is given in Sections 3.2 to 3.5.

2.3 The sampling scheme for Waverider buoys which is used at IOS is as follows:

A wave record consists of 2048 measurements of the buoy heave taken at intervals of 0.5 seconds. A record is taken every 3 hours, at midnight, 0300, 0600, etc.

It is considered that the record length presently used (which is limited by recording and analysis capacity) is too short and leads to rather high sampling errors. Record lengths of 2048 seconds (ie double our present practice) would give a useful improvement.

3. PROCESSING PROCEDURES AND PARAMETERS

3.1 This section contains an account of the methods used to calculate the spectrum of a sample record. We start by forming the Fourier series representation of the record. This is normally done on a computer using a Fast Fourier Transform technique, but, of course, the results are closely related to the well-known Fourier and Euler formulas:

$$h(t) = a_0 + \sum_{n=1}^{\infty} \left\{ a_n \cos \frac{2\pi n t}{T} + b_n \sin \frac{2\pi n t}{T} \right\} \quad (3.1)$$

$$a_0 = \frac{1}{T} \int_0^T h(t)\, dt \quad (3.2)$$

$$a_n = \frac{2}{T} \int_0^T h(t) \cos \frac{2\pi n t}{T}\, dt \quad (3.3)$$

$$b_n = \frac{2}{T} \int_0^T h(t) \sin \frac{2\pi n t}{T}\, dt \quad (3.4)$$

Note that a_0 is just the mean of the record. In view of the nature of the model we have adopted for the sea surface this must be zero, and we can impose this condition by subtracting the record mean from $h(t)$ before we Fourier analyse it.

We should also note that the total variance of the record is equal to the sum of the variances associated with each term in the Fourier series, ie:

$$Var(h(t)) = \sum_{n=1}^{\infty} \frac{1}{2} (a_n^2 + b_n^2) \quad (3.5)$$

and moreover, the variance of the record in a certain frequency band is just the sum of the variances of the corresponding Fourier components.

The Fourier components a_n and b_n produced by the FFT program are available at the fundamental frequency

$$f_0 = \frac{1}{T} = \frac{1}{N\Delta T} \quad (3.6)$$

and integral multiples thereof up to the Nyquist frequency, $1/2\Delta T$

3.1.1 Forming the sample spectrum

Now let us return to the model of Sec 1. and acknowledge that our best estimate of the variance of the process in any given frequency range is given by the variance in that range observed in the sample record, ie:

$$\sum_{n=i-j}^{i+j} \hat{C}_n^2 = \sum_{n=i-j}^{i+j} \tfrac{1}{2}(a_n^2 + b_n^2)$$

is our best estimate of $\int_{f=(i-j-\frac{1}{2})f_0}^{f=(i+j+\frac{1}{2})f_0} S(f)\,df$

Over a small frequency band we may approximate the integral in the following way:

$$\int_{f=(i-j-\frac{1}{2})f_0}^{f=(i+j+\frac{1}{2})f_0} S(f)\,df \simeq S(if_0)\cdot(2j+1)f_0$$

For a single component we thus have that:

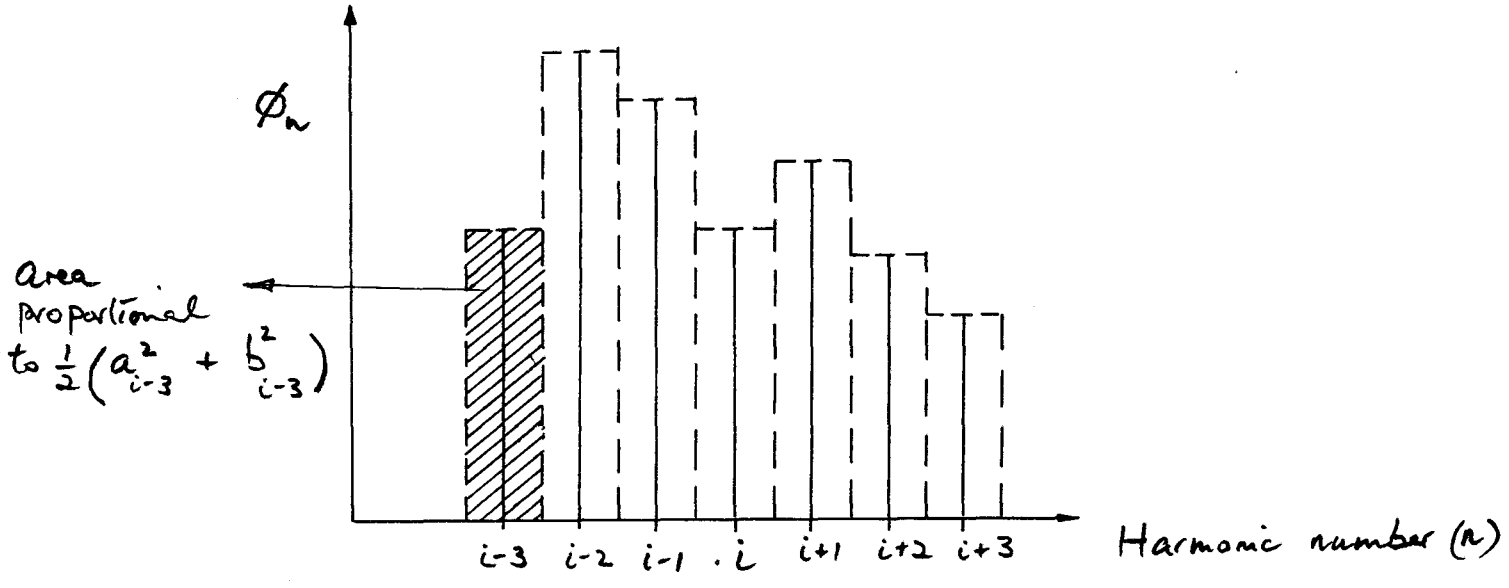
$\frac{1}{2}(a_i^2 + b_i^2)$ is our best estimate of $S(if_0)f_0$

ie $\frac{1}{2f_0}(a_i^2 + b_i^2)$ is our best estimate of $S(if_0)$

We define this quantity as the elementary sample estimate of the spectrum at frequency if_0, and write:

$$\frac{1}{2f_0}(a_n^2 + b_n^2) = \phi_n \qquad (3.7)$$

The diagram may make the reasoning clearer



Areas proportional to $\frac{1}{2}(a_n^2 + b_n^2)$ are drawn centred on their appropriate harmonic numbers. The sum of these areas is proportional to the variance over the range, (3.5).

Now, the areas can also be expressed as their individual heights multiplied by their individual widths; we call the height the sample estimate of the spectral density ie:

$$\sum \tfrac{1}{2}(a_n^2 + b_n^2) = \sum \phi_n f_0 \qquad (3.8)$$

where we have written ϕ_n for $\phi(nf_0)$

Finally, for a single estimate:

$$\phi_n = \frac{1}{2f_0}(a_n^2 + b_n^2) = \frac{T}{2}(a_n^2 + b_n^2) \qquad (3.9)$$

3.2 Smoothing of the Spectral Estimates

It can be shown that the sample estimate defined in (3.7) is subject to a very large random variability; its standard error is, in fact, 100%. In order to reduce this to acceptable proportions, it is usual to perform some kind of smoothing in the time and/or frequency domains.

In frequency-domain averaging, the record is analysed and the average of ℓ adjacent spectral estimates is taken as the final estimate, and ascribed to the mid-frequency of the band of estimates, thus

$$\left.\begin{aligned} S_i(f_i) &= \frac{T}{2\ell}\sum_{n=j}^{j+\ell-1}(a_n^2 + b_n^2) \\ \text{where} \quad j &= (i-1)\ell + 1 \\ \text{and} \quad f_i &= \frac{1}{T}(j + (\ell-1)/2) \end{aligned}\right\} \qquad (3.10)$$

In time-domain averaging the wave record is divided into q sub-sections of equal length. Each section is analysed and corresponding estimates from each of the q sub-sections are averaged to give the final estimate.

Thus

$$S_i(f_n) = \frac{1}{q}\sum_{k=1}^{q}\phi_k(f_n) \qquad (3.11)$$

where ϕ_k is the estimate calculated from the k th subsection,

and f_n is the frequency of the n th harmonic of q/T

The following parameters of these calculations are approximately correct (see Bendat and Piersol, 1971). For frequency domain averaging, the effective bandwidth is given by

$$\Delta f = \frac{l}{T} \tag{3.12}$$

The number of degrees of freedom is given by

$$\nu = 2l \tag{3.13}$$

The normalised standard error is given by

$$\epsilon_r = \sqrt{1/l} \tag{3.14}$$

(Substitution of (3.12) into (3.14) leads to (2.2)).

The corresponding parameters for time domain averaging are:

$$\left.\begin{aligned} \Delta f' &= q/T \\ \nu' &= 2q \\ \epsilon'_r &= \sqrt{1/q} \end{aligned}\right\} \tag{3.15}$$

3.3 Example Spectrum analysis design

The design of the analysis will depend upon the application, and will involve a compromise between the required resolution, and the required statistical stability. However, let us assume that we have at our disposal a wave record of 2048 points, taken at 0.5 second intervals. It has been found that a reasonable choice for the resolution is

$$\Delta f = 0.01 \text{ Hz}$$

Then from (3.12) l = 1024 x 0.01, selecting the nearest integer gives

$$l = 10$$

which in turn gives the exact value of the resolution as

$$\Delta f = 10/1024 = 0.009766 \text{ Hz}$$

The number of degrees of freedom of each estimate

$$\nu = 20$$

The normalised standard error is

$$\epsilon_r = \sqrt{1/10} = 32\%$$

The total number of elementary Fourier components will be

$$N\Delta T\left(\frac{1}{2\Delta T} - \frac{1}{N\Delta T}\right) + 1$$

(The frequency of the last estimate minus the frequency of the first, divided by the frequency separation, plus one).

$$= N/2 = 1024$$

The total number of smoothed estimates will be $1024/\ell = 102$ (next smaller integer).

These will occur at frequencies given by (3.10). In particular the frequency of the first estimate will be

$$f_1 = \frac{1}{1024}\left(1 + \tfrac{9}{2}\right) = 0.00537 \text{ Hz}$$

and of the last

$$f_{102} = \frac{1}{1024}\left(1011 + \tfrac{9}{2}\right) = 0.992 \text{ Hz}$$

3.4 Confidence limits for the spectral estimates

As well as defining the standard error of each estimate we can obtain information about the distribution of the errors, which in turn allows us to calculate confidence limits. For the example of Sec 3.3 these turn out to be rather large.

The spectral estimates are taken to be distributed as Chi-squared variables with ν degrees of freedom. In our example with $\nu = 20$, the 95% confidence limits are 2.11 and 0.58. That is, 19 times out of 20 on average, the true value of the spectrum can be expected to lie within 211% and 58% of the estimated value. If we were to double the sample length keeping Δf the same, we would double ν to 40, and the 95% confidence limits would then be 1.6 and 0.76.

3.5 Tapering of the Data Series

The frequencies at which the Fourier components are evaluated are fixed by the record length and will not in general coincide with the frequencies in the underlying process at which the variance is located. This gives rise to a phenomenon called 'leakage' in which the variance associated with a particular frequency in the underlying process leaks into a band of harmonics in the Fourier series of the sample. It can

be shown that the effect is negligible for most practical purposes in the example quoted above, but with shorter records it may in some circumstances be significant. For this reason the time-domain averaging method described above (in which the original record is sub-divided into shorter sections) should be approached with some care.

Leakage can in almost all cases be reduced to insignificant proportions by a process called 'tapering' in which the original record is multiplied by a function which increases smoothly from zero at each end of the record to unity in the central part.

Often, the so-called cosine taper is used which has the form:

$$\frac{1}{2}\left(1 - \cos\frac{2\pi i}{N}\right) \qquad (3.16)$$

where i is the sample point serial number and N is the total number of points. However, the use of a taper reduces the information content of the record and leads to higher sampling errors. The full cosine taper (3.16) gives a 40% increase in the standard error of the estimates, so that in the example of Sec 3.3 ϵ_r would increase from 32% to 45%.

A good compromise is to use a function of the form of (3.16) but applied to perhaps 12½% of the record at each end. This results in a substantial reduction in leakage (if this is thought to be necessary) while having only a small effect on the statistics of the calculation.

3.5.1 Restoration of Total Variance when a Taper Function has been used

Because of the random nature of the process, the application of a known taper function will reduce the variance of the record by an unpredictable amount. Now the basic definitions (1.1, 1.2, 3.5 and 3.7) imply that the total variance of the record must be equal to the integral of the spectrum over frequency. We expect the use of the taper to change the shape of the spectrum by a greater or lesser amount, but it is usual to require the total variance of the sample spectrum to be the same as the variance of the sample record. We impose this condition by scaling the final spectrum by an appropriate amount.

3.6 Integrated properties of the spectrum

The n th moment of the spectrum is defined as

$$m_n = \int_0^\infty S(f) f^n \, df \qquad (3.17)$$

In practice the integral is replaced by the finite sum

$$m_n = \sum_{i=j}^{k} S_i f_i^n \Delta f \qquad (3.18)$$

The selection of j and k should be undertaken with care.

It is not sensible to include in the sums contributions from frequency bands in which the performance of the instrument is known to be questionable. Moreover, for most engineering purposes the frequency band of interest is contained within the range 0.05Hz to 0.5Hz.

At IOS, in the case of Waverider measurements, the sums are formed over a frequency range 0.04Hz to 0.63Hz. Using (3.10) these correspond (in the example of Sec 3.3) to smoothed estimate numbers of 5 and 65.

The moments can also be calculated from the elementary Fourier components, with suitable adjustments to (3.18), thus

$$m_m = \sum_{n=j'}^{k'} \phi_n f_n^m f_0 \qquad (3.19)$$

The values of j' and k' which correspond to j and k above are 41 and 650.

Several wave parameters of interest can be calculated from the moments; in particular:

The significant waveheight, $H_s = 4\sqrt{m_0}$

The mean zero crossing period, Tz = $\sqrt{\frac{m_0}{m_2}}$

A number of additional period parameters can be calculated from the moments, the sequence

$$T(0,n)_{n\neq 0} = \left(\frac{m_0}{m_n}\right)^{\frac{1}{n}}$$

is of interest, in particular:

$T(0,2)$ = Tz, the zero crossing period

$T(0,1)$ = $\bar{T}$ the mean period

$T(0,-1)$ = T_e the energy period

3.6.1 Spectral peakedness parameters

Several spectral peakedness parameters have been proposed, including one by Goda which he calls Q_p, given by

$$Q_p = \frac{2\int_0^\infty f S^2(f)\,df}{m_0^2} \qquad (3.20)$$

The appearance of S^2 in the numerator makes Q_p sensitive to the degree of smoothing employed. In our example, values calculated from the unsmoothed spectrum are typically twice those calculated from the smoothed spectrum.

At IOS Q_p is calculated from the smoothed spectrum, thus:

$$Q_p = \frac{2\Delta f}{m_0^2} \sum_{i=5}^{65} f_i S_i^2 \qquad (3.21)$$

4. SEQUENCE OF OPERATIONS TO ESTIMATE THE SPECTRUM OF A WAVE RECORD

(1) Subject digital record to quality control scrutiny (not discussed in this note)

(2) Subtract mean of record

(3) Calculate variance of record

(4) Apply taper function if required

(5) Input data to FFT program

(6) Form spectrum as $T/2\,(a_i^2 + b_i^2)$

(7) Scale spectrum with instrument scaling etc (NB: ensure scaling applied in FFT algorithm is allowed for)

(8) Compare total variance of spectrum with variance of sample record This is always a useful check since it will reveal the scaling applied in the FFT program

(8(a)) If a taper function has been used, ensure that total variance of the spectrum is equal to the variance of the sample record by scaling the spectrum appropriately.

(9) Apply frequency response corrections for instrument, electronics, recorder etc.

(10) Calculate moments of the spectrum

(11) Average over adjacent harmonics to form final smoothed spectrum.

Note: This averaging process can be placed before item (9) with negligible effect on the accuracy in most instances. In any case the spectrum should be smoothed before the peakedness parameter is calculated.

(12) Calculate Q_p or other peakedness parameter.

5. QUALITY INDICATORS

At IOS the bulk of the automatic detection of errors is done using the (time domain) wave records themselves and the results of these checks are appended to the record as a series of flags. These flags are copied into the output of the spectral analysis.

In addition, it may be useful to calculate some quantities from the spectra which give an indication of quality. Experience has shown that such quantities can rarely be used 'unseen' as discriminators between usable and unusable spectra, but they are valuable as indicators as to whether or not the spectra should be examined further.

BIBLIOGRAPHY

BENDAT, J.A. and PIERSOL, A.G. (1971). Random data: Analysis and measurement procedures. Wiley-Interscience, New York.

BLACKMAN, R.B. and TUKEY, J.W. (1958). The measurement of Power Spectra. Dover Publications, New York.

BRIGHAM, O.E. (1974). The Fast Fourier Transform. Prentice Hall, Englewood Cliffs, N.J.

CARTWRIGHT, D.E. and LONGUET-HIGGINS, M.S. (1956). The statistical distribution of the maxima of a random function. Proceedings of the Royal Society of London. A237, 212-232.

GODA, Y. (1970). Numerical Experiments with Spectral Simulation. Port and Harbour Research Institute. Report 9, No.3.

GODA, Y. (1974). Estimation of Wave Statistics from Spectral Information. Pp 235-253 in Proceedings of the International Symposium on Ocean Wave Measurement and Analysis, Waves 74. Volume 2. New York: American Society of Civil Engineers.

JENKINS, E.M. and WATTS, D.E. (1968). Spectral analysis and its applications. Holden Day, San Francisco.

KINSMAN, B. (1965). Wind waves, their generation and progagation on the ocean surfaces. Prentice Hall, Englewood Cliffs, NJ.

APPENDIX B

Instructions for running the Spectral Analysis Program

Assignment of files:

The "run data", ie data which define the parameters to be used in each particular run of the program are input on Logical Unit Number (LUN) 1.

The input digital wave data files should be assigned in chronologically increasing order to LUN 1Ø and upwards. The only constraint on the number of files imposed by the program is that there should be 99 or less.

The main spectral file is written to LUN2.

The index file is written to LUN5

The size of the output files can be calculated using the following information.

The spectral files are made up of records with a uniform length of 96 characters. There are 4 file header records, and each spectrum occupies 18 records. "No-data" entries occupy 1 record.

The index files are similarly made up of 96 character records. There are 11 file header records. Each "data" entry occupies 2 records, "no-data" entries occupy 1 record.

Memory requirements: The program requires a minimum of 32K words of memory to run successfullly.

Entry of "run data": It will be found useful if a pro-forma is prepared which can be completed before each run of the program. A copy of the form used at IOS is included as figure B1.

First line (4ØA1)

ITITL — Title of data. This is written to both output files as the first file header record.

Second line (2A1)

ISITE — Site code.

INST — Instrument code.

Third line (I2, I3, 2I2)

ISTY — Last 2 digits of year (nominal) in which output data series starts.

ITD — Day number (nominal) on which output data series starts.

ITH — Nominal time (hours) at which output data series starts

ITM — Nominal time (minutes) at which output data series starts.

Fourth line (2I2, I3, I1)

INITM — Month number (nominal) in which output data series starts.

NIMF — Number of input files.

NDLY — Number of days in last year. This is used in the record allocation logic. "Last year" refers to the year preceding the last year-boundary whether or not the boundary is included in the output data period.

NPR — Taper scale.

Fifth line (I3, A1)

LAST — Day number of last day to be included in the output data series.

LOF — Type of logical end of file record (LEOF) to be written.
If "M" the standard LEOF is written (all nines).
If "S" the end of series LEOF is written (all eights) and provision is made for the input of a second scale factor.

Sixth line (I2)

NOR1 — Number of records whose status flag (KFLAG(9)) is to be specified as (or changed to) Ø. If NOR1 equals Ø, the next line is omitted.

Seventh line (NOR1 (I2, I2, I3))

Consists of NOR1 groups of three numbers, each group identifies one of the NOR1 records whose status flag is to be set to Ø. For the Nth such record the three numbers are as follows:

OR1TØ (N, 1) — Year of Nth record.

OR1TØ (N, 2) — Number of the input file in which the Nth record appears (equal to LUN-9).

OR1TØ (N, 3) — Observation number of Nth record as it appears in the input file.

Eighth line (I2)

NOR2 — Number of records whose status flag (KFLAG(9)) is to be specified as (or changed to) 1. If NOR2 is equal to Ø, the next line is omitted.

Ninth line (NOR2 (I2, I2, I3))

Consists of NOR2 groups of three numbers, each group identifies one of the NOR2 records whose status flag is to be set to 1. For the Mth such record the three numbers are as follows:

ORØT1	(M, 1)	Year of Mth record.
ORØT1	(M, 2)	Number of the input file in which the Mth record appears (equal to LUN-9).
ORØT1	(M, 3)	Observation number of the Mth record as it appears in the input file.

Tenth line (3F10.8)

SCALF	Scale factor for converting the digital units of the input data into users units (usually metres).
SINT	Sampling interval (seconds).
SCALF2	Second scale factor (if required).

Eleventh line (I3, 2I2)

This gives information on the precise date and time from which the second scale factor is to apply. The program assumes that a second scale factor will be required only if the LEOF is set to 'end of series' (LOF = 'S'). Otherwise this line is omitted.

ICSFD	Day number of scale factor change.
ICSFH	Hour of scale factor change.
ICSFM	Minute of scale factor change.

SPECTRAL ANALYSIS PROGRAM INPUT DATA

AS:
AS:

PROGRAM:

DATE RUN:
S NUMB:
FILES USED:

Title.

Site, Instrument code.

Nominal year, Dayno, Time of first data record.

Starting month number, Number of files to be processed, Number of days in previous year, Taper scale.

Last dayno in each month, LEOF mark (M or S).

Number of ϕ to 1 record status overides.

Year, Input file number, Record number of each overide (omit this line if none).

Number of 1 to ϕ record status overides.

Year, Input file number, Record number of each overide (omit this line if none).

Scale factor, Sampling int. Scale Factor 2 DayNo Time S.F. Change

Figure B1

APPENDIX C

Description of SUBROUTINE READ, and discussion of compatibility requirements of user replacement

SUBROUTINE READ (LUN, I, IW, KFLAG, LPAR, M, N, NEND)

Function

The subroutine reads from logical unit number LUN in one of two modes controlled by argument N.

If N = 1 the subroutine reads one wave observation consisting of an observation header and 2088 data points. The data are arranged in 36 blocks of 58 points, and appended to each block is a group of 10 error flags and a parity result character.

If N = 2 a four-record file header is read. Thus, at the start of an input file the subroutine is called once in mode N = 2, and then as many times as required in mode N = 1.

Explanation of arguments

LUN:Logical unit number from which data are to be read.

I:Observation identification array:

(1) Logger number
(2) Field tape block number
(3) Year
(4) Day
(5) Hour
(6) Minute
(7) Observation number in field tape.

Of these, only (4), (5) and (6) are used in the main program.

KFLAG (1,1) to KFLAG (1,1Ø):a group of 1Ø observation error flags which take values of 1 or Ø. If they are 'set' ie equal to 1, they have the following meanings:

(1, 1) Observation has been manually edited.
(1, 2) Digital data from digitised analogue observation.
(1, 3) Ship steaming (SBWR observation only)
(1, 4) Not used.
(1, 5) Observation start not an intermediate synoptic hour.
(1, 6) Inter-observation gap not 3 hours.
(1, 7) Not used.
(1, 8) Not used.

(1,9) Observation invalid.

(1, 1Ø) Observation length incorrect.

I and KFLAG (1, 1) to KFLAG (1, 1Ø) together constitute the observation header

IW 2Ø88 element array containing the data points - only the first 2Ø48 elements in IW are used in the main program

KFLAG (2, 1) to KFLAG (37, 1Ø)
Block error flags. These are not used in the main program.

LPAR 36 element array containing block parity result characters. Not used in main program.

IW, KFLAG (2, 1) to KFLAG (37, 1Ø) and LPAR together constitute the data

M (4, 4Ø) Array containing 4 line header. Not used in main program.

N Mode of call

NEND Returned as 1 if end of file encountered, otherwise Ø.

Requirements of replacement READ

In order to be compatible with the main program, a replacement read must return the following:-

I (4) Day number

I (5) Hour

I (6) Minute

I (7) Observation number in field tape

KFLAG (1, 1) to KFLAG (1, 1Ø) observation error flag array.

IW (1) to IW (2Ø48) 2Ø48 data points.

Mode of call - N

The main program makes one call in mode N = 2 at the beginning of each input file, followed by as many mode N = 1 calls as required.

APPENDIX D

Description of subprograms in WARP4S

INTRODUCTION

The subprograms contained in WARP4S are all concerned with the spectral analysis of wave data. They were written over a period of years to perform various analysis functions, and sometimes the same function but in slightly different ways. One constant factor during their development was that the SUBROUTINES FFT and RTRAN have been used to perform the Fourier analysis, and so all the subprograms are compatible with the input and output of FFT and RTRAN.

Descriptions of these two subprograms appear at Appendix G, and it should be born in mind when studying this that they are used in the present system of programs to calculate the single sided transform of a real data series (sample).

Scaling of FFT results:

Consider the finite discrete Fourier series representation of the (zero-mean) function h(m) defined over the finite interval 1(1)N

$$h(m) = \sum_{n=1}^{N/2} \left\{ a_n \cos \frac{2\pi n m}{N} + b_n \sin \frac{2\pi n m}{N} \right\} \qquad -1$$

After FFT and RTRAN have been called, the elements of arrays A and B bear the following relationship to the Fourier coefficients a and b in equation 1 (see appendix A):

$$\begin{bmatrix} a_L \\ b_L \end{bmatrix} = \begin{bmatrix} A(L+1)/N \\ B(L+1)/N \end{bmatrix} \qquad -2$$

In 1 and 2 N is the number of points in the data series. In the program system this is represented by MOBS.

The elementary sample estimate of the spectrum (the periodogram) is defined as follows (see appendix A).

$$S(L) = {}^{T}/_{2}(a_L^2 + b_L^2) \qquad -3$$

where T is the series length in seconds, and equals MOBS x SINT, SINT being the sampling interval in seconds.

$$\text{Thus } S(L) = \frac{\text{MOBS x SINT}}{2} \quad (a_L^2 + b_L^2) \qquad -4$$

which with (2) gives

$$S(L) = \frac{\text{SINT}}{\text{MOBS x 2}} \quad (A^2\ (L + 1) + B^2\ \ (L + 1)) \qquad -5$$

A list of the subprograms which are available in WARP4S is now given in the order in which they appear in the file. In this order, they constitute a subprogram library.

Interdependence of subprograms:

Not all of the subprograms are independent of each other. The application of a cosine taper, for example, requires an initialisation call to

SUBROUTINE SINTAB followed by a call to SUBROUTINE DATA2, which prepares the data for FFT while applying a cosine taper. Within DATA2, FUNCTION QSIN2 is used to look-up the value of the taper function, the two programs being completely interdependent.

Each subprogram description gives information on its dependence on other subprograms in the system.

Array indexing convention

In order to be consistent with the output of RTRAN which, according to equation 2, puts the first harmonic in element 2 of the transform arrays, the second in element 3 and so on, all tabulations relating to the Fourier transform including frequency calculations follow this same convention.

However, when frequency domain averaging has been performed the first smoothed estimate above zero frequency is placed in element 1 of the relevant array and this also applies to the associated frequencies.

To avoid confusion, the above information is repeated at all relevant points in the individual subroutine documentation.

Naming conventions of variables

The names of the dummy arguments used in the subprograms are, as far as possible, consistent throughout the system. However, there are exceptions to this rule and the documentation of each subprogram should be carefully studied before calls are coded.

SUBROUTINE FDAV (NN, NAV, NOL, A, B, MOBS, NE, SCALF, SINT, NIA, NIB)
(Frequency-domain averaging)

Purpose

Starting with the discrete Fourier transform of a time series in arrays A and B, FDAV does the following:

(1) Forms the spectrum (see equation 5).

(2) Averages over NAV adjacent elementary (periodogram) estimates, with overlap NOL.

(3) Applies a scaling factor.

(4) Forms a table of frequencies for the averaged spectrum.

Argument list

NN	Number of components in discrete Fourier transform (= MOBS/2).
NAV	Number of adjacent elementary spectrum estimates to be included in each smoothed spectral estimate.
NOL	Number of elementary estimates to be common to adjacent smoothed estimates. If no overlap required, set to zero.
A	Array which initially contains the cosine terms of the discrete Fourier transform (see equation 2). The array is returned containing the smoothed spectrum in elements A(1) to A(NE).
B	Array which initially contains the sine terms of the discrete Fourier transform. The array is returned containing the frequencies for the smoothed spectrum in elements B(1) to B(NE).
MOBS	Number of points in the data series.
NE	Number of smoothed spectral estimates in final spectrum.
SCALF	Scaling factor (linear) to convert wave heights to user units (usually to metres).
SINT	Sampling interval in seconds.
NIA	Dimension information for A.
NIB	Dimension information for B.

Dependence on other subprograms in the system

Calls to FFT and RTRAN to calculate the Fourier transform must precede a call to FDAV.

SUBROUTINE FLIM (TETOP, TBOTT, ISTAR, ISTOP, B, NIB)
(Frequency limit)

Purpose

FLIM inspects array B and identifies those elements which contain frequencies which are closest to $^{1}/_{TETOP}$ and $^{1}/_{TBOTT}$. The corresponding indices are returned as ISTAR and ISTOP. Used for defining the summation range for the calculation of moments etc.

Argument list

TETOP	Period in seconds of low frequency limit.
TBOTT	Period in seconds of high frequency limit.
ISTAR	Low-frequency summation limit.
ISTOP	High-frequency summation limit.
B	Array containing monotonically increasing table of frequencies.
NIB	Dimension information for B.

Dependence on other subprograms in the system

The frequency table, B, would normally be output from SUBROUTINE FDAV.

FUNCTION CCAL2 (FREQ, AK, D)

Purpose

To calculate the amplitude frequency response correction for the type 4856 ship-borne wave recorder.

Argument list

FREQ	Frequency in Hz. A low-frequency limit is included so that that if FREQ is less than 0.04 Hz, the correction appropriate to 0.04 Hz is returned.
AK	K value. The true sensor depth D is multiplied by this number before the correction is evaluated. Often set to 2.5.
D	Pressure sensor depth (metres).

Dependence on other subprograms in the system

Nil.

FUNCTION DCAL2 (FREQ)

Purpose

To calculate the amplitude frequency response correction for the Datawell Waverider buoy.

Argument list

FREQ	Frequency (Hz). NB A low-frequency limit is included so that if FREQ is less than 0.04 Hz, the correction appropriate to 0.04 Hz is returned.

Dependence on other subprograms in the system

Nil

FUNCTION FMOM (N, B, E, MOBS, ISTAR, ISTOP, NAV, SINT, NIB, NIE)
(Frequency moment)

Purpose

To calculate the frequency moment of order N of the spectrum.

Argument list

N	Order of moment to be calculated.
B	Array containing table of frequencies of the spectrum.
E	Array containing corresponding spectral estimates.
MOBS	Number of points in original data series.
ISTAR	Low frequency summation limit) These expressed as the number of
ISTOP	High frequency summation limit) the element of B containing the relevant frequency.
NAV	Number of elementary spectral estimates over which estimates in E have been averaged.
SINT	Sampling interval in seconds.
NIB	Dimension information for B.
NIE	Dimension information for E.

Dependence on other subprograms in the system

Calls to FFT and RTRAN and other appropriate subprograms must be made so that the spectrum is available in E, with its corresponding frequencies in B.

FUNCTION TOVAR (NSEC, RVAR, RMEAN, MOBS)
(Total variance)

Purpose

To calculate the total variance of a data series which has been divided into sections of equal length whose individual means and variances are known.

Argument list

NSEC	Number of sections into which data series is divided.
RVAR	Array containing the variances of the NSEC sections.
RMEAN	Array containing the means of the NSEC sections.
MOBS	Number of points in each section.

Dependence on other subprograms in the system

Nil

SUBROUTINE BSTAT (IX, MOBS, XMEAN, VAR)
(Basic Statistics)

Purpose

Calculate the mean and variance of the data series.

Argument list

IX	Array containing data points.
MOBS	Number of points in the series.
XMEAN	Mean of the series.
VAR	Variance of the series.

Dependence on other subprograms in the system

Nil

SUBROUTINE CQDAV (NN, NAV, NOL, A, B, C, D, MOBS, NIA, SINT, COSP, QUAD, FREQA, NE, NINE, SCALF)

(Development of FDAV to form co- and quad-spectra)

Purpose

Starting with the discrete Fourier transforms (DFT's) of two related time series in arrays A, B, C and D, CQDAV does the following:

(1) Forms the co-spectrum and the quadrature-spectrum of the series.

(2) Averages over NAV adjacent elementary estimates, with overlap NOL.

(3) Applies a scaling factor.

(4) Forms a table of frequencies for the averaged spectra.

Argument list

NN	Number of components in discrete Fourier transforms (= MOBS/2).
NAV	Number of adjacent elementary spectrum estimates to be included in each smoothed spectral estimate.
NOL	Number of elementary estimates to be common to adjacent smoothed estimates. If no overlap required, set to zero.
A	Array containing the cosine terms of the DFT of data series 1.
B	Array containing the sine terms of the DFT of data series 1.
C	Array containing the cosine terms of the DFT of data series 2.
D	Array containing the sine terms of the DFT of data series 2.
MOBS	Number of points in data series.
NIA	Dimension information for A, B, C and D.
SINT	Sampling interval in seconds.
COSP	Array containing co-spectrum.
QUAD	Array containing quad-spectrum.
FREQA	Array containing a table of frequencies for spectra.
NE	The number of spectral estimates in COSP and QUAD.
NINE	Dimension information for COSP, QUAD and FREQA.
SCALF	Scale factor (linear) for converting wave heights to user units.

Dependence on other subprograms in the system

Calls to FFT and RTRAN to form the DFT's, A, B and C, D are required before CQDAV is called.

FUNCTION QUPEE (B, E, MOBS, ISTAR, ISTOP, NAV, SINT, NIB, NIE)
(Goda's Qp)

Purpose

To calculate Goda's spectral peakedness parameter, Qp.

Argument list

B	Array containing list of frequencies of smoothed spectrum.	
E	Array containing smoothed spectrum.	
MOBS	Number of points in original data series.	
ISTAR	Low frequency summation limit)	These expressed as the number of the element of B containing the relevant frequency.
ISTOP	High frequency summation limit)	
NAV	Number of elementary spectral estimates over which estimates in E have been averaged.	
SINT	Sampling interval in seconds.	
NIB	Dimension information for B.	
NIE	Dimension information for E.	

Dependence on other subprograms in the system

The smoothed spectra and its frequency table must be formed by calling FDAV or FDAV2 before QUPEE is called.

SUBROUTINE DATA2 (A, B, IX, MOBS, NN, NIA, NIB, NIIX, ISILU, NPR)

Purpose

To transfer the data series contained in array IX into arrays A and B in the way required by FFT.

In addition a variable cosine taper can be applied to the data.

Argument list

A	Array containing prepared data.
B	Array containing prepared data.
IX	Array containing digital data series.
MOBS	Number of points in data series.
NN	Number of elementary components in discrete Fourier transform (= $^{MOBS}/_{2}$). Also called "order" of FFT.
NIA	Dimension information for A.
NIB	Dimension information for B.
NIIX	Dimension information for IX.
ISILU	Array containing look-up table of values for cosine taper.
NPR	Cosine taper is applied to $^{1}/_{NPR}$ of the data series at each end. Thus NPR = 2 results in a full cosine taper. If NPR = 0, DATA2 calls SUBROUTINE DATA and RETURNS.

Dependence on other subprograms in the system

If NPR = 0, SUBROUTINE DATA is called.

If NPR $\neq$ 0, FUNCTION QSIN2 is used to obtain the taper coefficients from ISILU, which must be initialised by a prior call to SINTAB.

SUBROUTINE DATA (A, B, IX, MOBS, NN, NIA, NIB, NIIX)

Purpose

To transfer the data contained in array IX into arrays A and B in the way required by FFT.

Argument list

A	Array containing prepared data.
B	Array containing prepared data.
IX	Array containing digital data series.
MOBS	Number of points in data series.
NN	Number of elementary components in discrete Fourier transform (= $^{MOBS}/_{2}$). Also called "order" of FFT.
NIA	Dimension information for A.
NIB	Dimension information for B.
NIIX	Dimension information for IX.

Dependence on other subprograms in the system

Nil

SUBROUTINE SINTAB (ISILU, NPR, MOBS)
(Sine-squared tabulation)

Purpose

To initialise table of coefficient for use in applying a cosine-taper.

Argument list

ISILU	Array containing the table of taper coefficients (Internally dimensioned as 512).
NPR	Cosine taper is applied to $^1/_{NPR}$ of the data series at each end. Thus NPR = 2 results in a full cosine taper. If NPR = 0, the subprogram returns immediately, and no attempt is made to initialise ISILU.
MOBS	Number of points in data series.

Dependence on other subprograms in the system

The form in which the taper is storedin ISILU is highly specific and requires the use of DATA2 and QSIN2 to access it.

FUNCTION QSIN2 (II, NANTA, ISILU)
(Quick sine-squared)

Purpose

Used in conjunction with DATA2 to derive the taper coefficients from ISILU.

Argument list

II	Order of coefficient.
NANTA	Number of data points at each end of series to be tapered.
ISILU	Array containing the table of taper coefficients. (Internally dimensioned as 512).

Dependence on other subprograms in the system

FUNCTION QSIN2 is called by DATA2 to apply a cosine taper to the data. It should not be used indepently of DATA2.

SUBROUTINE FTAB (MOBS, NN, SINT, FLUP, NAV, NOL, FAVTAB)
(Frequency table)

Purpose

To initialise tables of the frequencies corresponding to the elementary discrete Fourier transform (periodogram) of the data, and the smoothed (averaged) spectrum.

Argument list

MOBS	Number of points in data series.
NN	Returned as MOBS/2.
SINT	Sampling interval (seconds).
FLUP	Table of frequencies (Hz) for the elementary spectral estimates (periodogram). Note that the fundamental frequency is placed in element 2, the second harmonic in element 3 etc.
NAV	Number of elementary spectral estimates included in each smoothed spectral estimate.
NOL	Number of elementary estimates which are common to adjacent smoothed estimates. If no overlap required, set to zero.
FAVTAB	Table of frequencies for the smoothed spectral estimates.

Dependence on other subprograms in the system

Nil.

SUBROUTINE FCORR (NN, FLUP, FUNC1, FUNC2, CORLUP)
(Frequency dependent correction)

Purpose

To initialise a table of frequency-dependent corrections to be applied to the spectrum. Tabulated corrections are the squares of the products of two functions of frequency.

Argument list

NN	Number of elementary components in discrete Fourier transform (= MOBS/2). Also called "order" of FFT.
FLUP	Table of frequencies (Hz) for the elementary spectral estimates (periodogram).
FUNC1	FUNCTION subprogram of frequency which defines form of first amplitude correction.
FUNC2	FUNCTION subprogram of frequency which defines form of second amplitude correction.
CORLUP	Array containing correction factors corresponding to the frequencies in FLUP, ie first factor is placed in element 2. At each frequency the correction is formed as the square of the product of the two amplitude correction functions ie it is an "intensity" correction.

Dependence on other subprograms in the system

The look-up table of frequencies, FLUP, must be initialised by a call to FTAB before FCORR is called.

The FUNCTION subprograms FUNC1 and FUNC2 must be declared EXTERNAL in the calling program.

SUBROUTINE SPEC (A, B, NN, MOBS, CORLUP, SINT, SCALF, VAR, SCADJ2)
(Spectrum)

Purpose

Starting with the discrete Fourier transform of a time series in arrays A and B, SPEC does the following:

(1) Calculates the taper adjustment factor SCADJ2.

(2) Forms the unsmoothed spectrum (periodogram) of the data series. (Equation 5).

(3) Applies frequency-dependent corrections, taper adjustment and scale factors.

Argument list

A	Array which initially contains the cosine terms of the discrete Fourier transform (equation 2). The array is returned containing the unsmoothed spectrum (periodogram) of the data series in elements 2 to (NN + 1).
B	Array which contains the sine terms of the discrete Fourier transform (equation 2).
NN	Number of points in the data series.
CORLUP	Array containing table of frequency-dependent corrections to be applied to the periodogram.
SINT	Sampling interval (seconds).
SCALF	Scale factor (linear) to convert wave heights to user units (usually to metres).
VAR	Variance of data series.
SCADJ2	Taper adjustment factor.

Dependence on other subroutines in the system

Calls to FFT and RTRAN to calculate the Fourier transform of the data series must precede a call to SPEC.

CORLUP must be initialised by a call to FCORR.

VAR may be calculated using BSTAT.

SUBROUTINE FDAV2 (A, E, NAV, NOL, NE, NN)

(Frequency-domain averaging, 2nd method)

Purpose

Starting with the unsmoothed spectrum (periodogram) of the data series, FDAV2 forms the average over NAV adjacent harmonics with an overlap of NOL.

Argument list

A	Array which initially contains the periodogram of the data series.
E	Array which is returned containing the smoothed spectrum.
NAV	Number of adjacent elementary spectrum estimates to be included in each smoothed spectral estimate.
NOL	Number of elementary estimates to be common to adjacent smoothed estimates. If no overlap required, set to zero.
NE	Number of smoothed spectral estimates in final spectrum.
NN	Number of components in the periodogram.

Dependence on other subprograms in the system

A call to SPEC to form the periodogram of the record must precede the call to FDAV2.

FUNCTION UNITY (FREQ)

Dummy function of frequency

Purpose

For use with SUBROUTINE FCORR when only one of the two FUNCTIONS is required, the other is set to UNITY, which always returns the result 1.0.

Argument list

FREQ	Frequency in Hz.

Dependence on other subprograms in the system.

Nil.

FUNCTION WRAMP (FREQ)
(Waverider amplitude)

Purpose

To calculate the amplitude frequency response correction for the Datawell Waverider buoy.

Argument list

FREQ	Frequency (Hz). NB A low frequency limit is included so that if FREQ is less than 0.04 Hz, the correction appropriate to 0.04 Hz is returned.

Dependence on other subprograms in the system

WRAMP is identical in all respects with DCAL2.

SUBROUTINE CQSPEC (A, B, AMPLUP, CO, QUAD, SBSPEC, E, SCALF, SINT, SCADJ1, SCADJ2, MOBS, NN, VAR, IND, NIA)
(Co- and quad-spectrum)

Purpose

CQSPEC calculates the standard spectral quantities obtainable from two related data series. In addition and in particular, it can combine the pressure and heave information from the Shipborne Wave Recorder (SBWR) to give the total spectrum. Starting with the discrete Fourier transforms of the two data series CQSPEC does the following:

(1) Calculates and applies the taper adjustment factor SCADJ1 for each series.

(2) Applies the amplitude correction functions to the Fourier coefficients of the two series.

(3) Applies the necessary scaling, including instrument scaling to the two series. Further functions of the subroutine are controlled by the mode indicator IND:-

(4) If IND equals 0, the subroutine forms the SBWR spectrum and returns.

If IND equals 1, the subroutine does the following:-

(5) Forms the two auto-spectra.

(6) Forms the quadrature spectrum.

(7) Forms the co-spectrum.

If IND is greater than 1, the subroutine carries out all functions (1) to (7).

Argument list

A	2-dimensional array containing the cosine terms of the discrete Fourier transforms of the two data series.
B	2-dimensional array containing the sine terms of the discrete Fourier transforms of the two data series.
AMPLUP	2-dimensional array containing the amplitude response corrections for the two Fourier transforms. These would normally be those appropriate to the pressure and heave channels of the type 5254 SBWR.
CO	Array containing unsmoothed estimate of the co-spectrum of the two data series.
QUAD	Array containing unsmoothed estimate of the quad-spectrum of the two data series.
SBSPEC	Array containing the unsmoothed estimate of the combined SBWR

	spectrum.
E	2-dimensional array containing unsmoothed estimates of the two auto-spectra of the two data series.
SCALF	2-element array containing the scale factors (linear) required to convert the data series units to user units.
SINT	Sampling intervals (seconds).
SCADJ1	2-element array containing square-roots of SCADJ2.
SCADJ2	2-element array containing the taper adjustment factors of the two series.
MOBS	Number of points in the data series.
NN	Number of components in discrete Fourier transforms (= MOBS/2).
VAR	2-element array containing the variancesof the two data series.
IND	Mode of call.
NIA	Dimension information for the 2-dimensional arrays A, B, AMPLUP, and E, eg A is dimensioned A (NIA, 2).

Dependence on other subprograms in the system

Calls to FFT and RTRAN to form the discrete Fourier transforms of the two data series must preceed the call to CQSPEC.

The array AMPLUP must be initialised before CQSPEC is called. In the case of data from the type 5254 SBWR, this is done by calling SUBROUTINE SBTAB.

SUBROUTINE CQDAV2 (E, CO, QUAD, SBSPEC, GAMMA2, NAV, NOL, NN, NE, IND, NIA)
(Extension of FDAV2 to perform frequency domain averaging of co- and quad-spectra)

Purpose

Starting with unsmoothed estimates of the several spectral and cross-spectral quantities obtained from two data series CQDAV2 forms the average over NAV estimates with overlap NOL. The precise function of the subroutine is controlled by the mode indicator IND:

If IND equals 0, only the combined SBWR spectrum is averaged.

If IND equals, 1, averaged values of the two auto-spectra, the co- and the quad-spectra and the coherency are returned.

If IND is greater than 1, all those functions included under IND equals 0 and IND equals 1 are performed.

Argument list

E	2-dimensional array initially containing the unsmoothed estimates of the two auto-spectra. Returned containing the averaged auto-spectra. Note that the unsmoothed auto-spectrum of series 1 is contained in elements E (2, 1) to E (NN + 1, 1). The smoothed spectrum is contained in elements E (1, 1) to E (NE, 1).
CO	Array initially containing the unsmoothed estimate of the co-spectrum of the two series. Returned containing the averaged co-spectrum.
QUAD	Array initially containing the unsmoothed estimate of the quad-spectrum of the two series. Returned containing the averaged quad-spectrum.
SBSPEC	Array initially containing the unsmoothed estimate of the SBWR spectrum. Returned containing the averaged SBWR spectrum
GAMMA2	Contains the coherency (γ^2) of the two series.
NAV	Number of adjacent elementary spectrum estimates included in each smoothed estimate.
NOL	Number of elementary estimates to be common to adjacent smoothed estimates. If no overlap required, set to zero.
NN	Number of components in unsmoothed spectra.
NE	Number of smoothed estimates in final spectra.
IND	Mode of call.
NIA	Dimension information for E, ie E is dimensioned E (NIA, 2).

SUBROUTINE SBTAB (FLUP, AK, D, NN, AMPLUP,LFLIM, HFLIM, NIA)
(Tabulate SBWR corrections)

Purpose

SBTAB tabulates the amplitude frequency response corrections for the pressure channel and the heave channel for the 5254 SBWR.

Argument list

FLUP	Table of frequencies (Hz) for which corrections are to be evaluated.
AK	K-value. The true sensor depth D is multiplied by this number before the correction is evaluated. Often set to 2.5.
D	Pressure sensor depth (metres).
NN	Number of elementary spectral estimates for which correction required (= $^{MOBS}/_{2}$).
AMPLUP	2-dimensional array containing the amplitude response corrections for pressure channel and the heave channel.
LFLIM	Low-frequency limit (Hz). At frequencies below this the correction appropriate to LFLIM is returned. LFLIM is a REAL variable.
HFLIM	High-frequency limit (Hz). At frequencies above this the correction appropriate to HFLIM is returned.
NIA	Dimension information for AMPLUP ie AMPLUP is dimensioned (NIA, 2).

Dependence on other subprograms in the system

The frequency table FLUP must be initialised by a call to FTAB before SBTAB is called.

APPENDIX E

Example spectral analyses using the subroutines in WARP4S

Introduction

Study of Appendix A will reveal that the spectral analysis of wave records can be undertaken in a number of slightly different ways. The exact method selected will depend on the circumstances, although for most applications the difference between the results using the different methods here described are rather small.

The relevant analysis choices are as follows:-

(1) Data series may be tapered or not.

(2) The smoothed spectrum may be formed before

  (a) the application of frequency response corrections and

  (b) the calculations of the moments

  OR

  (c) the periodogram may be corrected for frequency response and

  (d) the moments calculated from the corrected periodogram, before finally

  (e) the periodogram is smoothed to produce the final spectrum.

Purpose of the examples

The examples are not complete working programs, but show the sequence in which the subprograms from WARP4S should be called to accomplish the following analysis:

Example 1

Calculate spectrum of a 2048-point data series sampled at 2 Hz without application of taper. Form smooth spectrum by averaging over 10 harmonics apply frequency response corrections; calculate moments of spectrum, calculate Qp. (This method was used in a small computer where memory was very restricted.)

Example 2

Calculate spectrum of 1024 point data series sampled at 1 Hz applying full cosine taper. Form periodogram; apply frequency response corrections; calculate moments of spectrum; smooth spectrum over 10 harmonics, calculate Qp.

Note on summation range of moments

In each example we will sum the moments over a frequency range of

(approximately) 0.05 to 0.5 Hz. The indices of the corresponding spectral estimates can be calculated for example using equation (3.18) in Appendix A. Alternatively, SUBROUTINE FLIM could be called at an appropriate point, but this method is not used in the examples.

In example 2 the situation is a little more involved. Qp is to be calculated from the smoothed spectrum, whereas the moments are to be calculated from the periodogram. For the sake of tidiness, if for no other reason, it is desirable that the estimates of all the integrated properties should be based on the same periodogram estimates. Thus, summation over 0.05 to 0.5 Hz requires that we sum over smoothed spectral estimate numbers 5 to 51. This range contains contributions from periodogram estimate numbers 41 to 510; but we should note that these estimates are contained in array elements 42 to 511, and these are the values of ISTAR and ISTOP which are used in the calculation of the moments.

EXAMPLE 1

```
      DIMENSION IX (2048), A (1025), B (1025), E (1025), RMOM (7)
      EQUIVALENCE(A (1), E (1)).
                        :
      Read 2048 data values into IX, read SCALF
                        :
      Calculate mean and variance of data
      CALL BSTAT (IX, 2048, XMEAN, VAR)
      Subtract mean
      DO 10 N = 1, 2048
10    IX (N) = IX (N) - IFIX (XMEAN)
      Prepare data for FFT
      CALL DATA (A, B, IX, 2048, NN, 1025, 1025, 2048)
      Compute Fourier Transform
      CALL FFT (A, B, NN, NN, NN, 1, 1025, 1025)
      CALL RTRAN (A, B, NN, 1, 1025, 1025)
      Form and smooth spectrum
      CALL FDAV (NN, 10, 0, A, B, 2048, NE, SCALF, 0.5, 1025, 1025)
      Apply frequency response corrections. Now that array A contains the
      spectrum we refer to it as E.
      DO 20 N = 1, NE
20    E (N) = E (N)* DCAL2 (B(N))**2
      Calculate 7 moments of order -2 to +4
```

```
      DO 3Ø M = 1, 7
      N = M-3
3Ø    FMOM (M) = FMOM (N, B, E, 2Ø48, 5, 5, 1Ø, Ø.5, 1Ø25, 1Ø25)
      Calculate Qp
      QP = QUPEE (B, E, 2Ø48, 5, 51, Ø.5, 1Ø25, 1Ø25)
```

EXAMPLE 2

```
      DIMENSION IX (1Ø24), A (513), B (513), E (513), RMOM (7), ISILU (512),
      FLUP (512), FAVTAB (51), CORLUP (512)
      EQUIVALENCE (A (1), E (1))
      EXTERNAL WRAMP, UNITY
      DATA NN/512/
      Initialise  taper look-up table
      CALL SINTAB (ISILU, 2, 1Ø24)
      Initialise tables of frequencies
      CALL FTAB (1Ø24, NN, 1.Ø, FLUP, 1Ø, Ø, FAVTAB)
      Initialise table of frequency-dependent corrections
      CALL FCORR (NN, FLUP, WRAMP, UNITY, CORLUP)
                          ¦
                          ¦
      Read 1Ø24 data values into IX, read SCALF
                          ¦
                          ¦
      Calculate mean and variance of data
      CALL BSTAT (IX, 1Ø24, XMEAN, VAR)
      Subtract mean
      DO 1Ø  N = 1, 1Ø24
1Ø    IX (N) = IX (N) - IFIX (XMEAN)
      Apply taper to data and prepare for FFT
      CALL DATA2 (A, B, IX, 1Ø24, NN, 513, 513, 1Ø24, ISILU, 2)
      Compute Fourier Transform
      CALL FFT (A, B, NN, NN, NN, 1, 513, 513)
      CALL RTRAN (A, B, NN, 1, 513, 513)
      Form periodogram, apply taper adjustment, frequency response corrections
      to scale factor
      CALL SPEC (A, B, NN, 1Ø24, CORLUP,  1.Ø, SCALF, VAR, SCADJ2)
      Calculate 7 moments of order -2 to +4
      DO 3Ø M = 1, 7
      N = M-3
```

```
3Ø   RMOM (M) = RMOM (N, FLUP, E, 1Ø24, 41, 51Ø, 1, 1.Ø, 513, 513)
     Average spectrum over 10 harmonics, without overlap
     CALL FDAV2 (A, E, 1Ø, Ø, NE, NN)
     Calculate Qp from smoothed spectrum
     CALL QUPEE (FAVTAB, E, 1Ø24, 5, 51, 1Ø, 1.Ø, 513, 513)
```

APPENDIX F

Format and content of spectral data files

A spectral file and an index file are written at each run of the program. They each consist of formatted records with a uniform length of 96 characters. The entry in the output file corresponding to one digital site wave recording will be called an 'observation'. Hereafter, 'record' will mean 'logical record'.

Spectral file structure

The file starts with a file header consisting of 3 records, these are followed by the observations and the file is terminated by a logical end of file record.

Format of file header

Record 1

Data header record FORMAT (5X, 40A1, 51X)

Record 2

ISITE	Site Code
INST	Instrument Code

FORMAT (2A1, 94X)

Record 3

ISTY	Nominal year of first observation
ITD	Nominal day number of first observation
ITH	Nominal time of first observation (Hours)
ITM	Nominal time of first observation (Minutes)

FORMAT (1X, I2, I3, 1X, 2I2, 84X)

Format of observation

Each observation consists of 18 records.

Each record consists of 96 characters.

These records contain the following information:

Record 1

ISTY	Last two digits of year
NDAY	Day number
IHR	Hour
MIN	Minutes

HS	Significant wave height
TZ	Mean zero-crossing period
SNRL	Quality figure, low frequencies
SNRH	Quality figure, high frequencies
SCADJ1	Taper adjustment factor
KFLAG	(1-10) Validation flags
NREC	Record number (address) of this record with respect to the beginning of the file

FORMAT (5X, I2, I3, 3X, 2I2, 5 (3X, F6.2) 2X, 1ØI1, 4Y, I5, 11X)

Record 2

RM (1-7)	Moments of spectrum order -2 to +4
QP	Goda's spectral peakedness parameter

FORMAT (8E12.5)

Records 3-18

Tabulation of FREQ (I) frequency (Hz) and
S (I) spectral density (m^2/Hz)
as 64 pairs, arranged 4 per record as follows:

FREQ (1)	S(1)	FREQ (17)	S(17)	- - - - - - - - - -	FREQ (49)	S(49)
FREQ (2)	S(2)	FREQ (18)	S(18)	- - - - - - - - - -	FREQ (50)	S(50)
⋮	⋮	⋮	⋮		⋮	⋮
FREQ (16)	S(16)	FREQ (32)	S(32)	- - - - - - - - - -	FREQ (64)	S(64)

This can be read using the following statement:

READ (n, 8ØØ) ((FREQ (N + M - 1), M = 1, 49, 16), N = 1, 16)
8ØØ FORMAT (4(3X, F6.4, 3X, E12.5))

'No data' record

Only Record 1 appears in a 'no-data' observation, and in this all real numbers are set to 99.99, and the validation flags are set to 9. NREC, the record address continues to increment correctly.

File termination

There is a logical end of file record which takes the place of Record 1 of the next observation after the end of the data: ISTY (LEOF)
NDAY (LEOF)

IHR (LEOF)

MIN (LEOF)

FORMAT (5X, I2, 2X, I3, 3X, 2I2, 77X)

Index file structure

Index files are a separate file series which contain a subset of the information contained in the spectral files. They were designed to be written to a line-printer and so form a convenient hard-copy index to the spectral files. However, since they contain a fairly complete parameterisation of the spectra in terms of their moments spectral peakedness and quality indicators they are often used as the main data source.

File header records

There are 11 file header records, as follows:-

Record 1	Fortran carriage control '1' (form feed)
Record 2	15X,'INSTITUTE OF OCEANOGRAPHIC SCIENCES'
Record 3	Blank
Record 4	5X, 'SPECTRAL ANALYSIS OF WAVE DATA'
Record 5	Blank
Record 6	25X, ITITL (40A1)
Record 7	Blank
Record 8	5X, 'YEAR', 2X, ISTY (I4), 5X, 'MONTH NUMBER', 2X, MON (I2)
Record 9	Blank
Record 10	5X, 'SITE', 5X, ISITE (A1), 5X, 'INSTRUMENT', 5X, INST (A1)
Record 11	Blank

Observations

Each observation is represented by the first 2 records of the spectral observation as they appear in the spectral file.

'No-data' observations appear as one-record entries as in the spectral file.

File termination

Each index file is terminated by the same logical end of file (LEOF) code as its corresponding spectral file.

APPENDIX G

Description of SUBROUTINES FFT and RTRAN

Appendix G consists of the standard IOS documentation for the library SUBROUTINES FFT and RTRAN. The versions of these subroutines which are used in the spectral analysis suite of programs differ from the library versions in that the dimensions of the main array variables are included as arguments of the call, thus:-

```
CALL FFT (A, B, N, N, N, ISN, NIA, NIB)
```

Where NIA and NIB are the respective dimensions of arrays A and B.

SUBROUTINE FFT(A,B,NTOT,N,NSPAN,ISN)

Function

Complex, multivariate, mixed-radix, Fast Fourier Transform.
(Data series length not restricted to 2**n)
N.I.O Subprogram -134 on IBM 1800 computer.

This Fast Fourier Transform subroutine has great advantages over previous ones in that the lengths of data series used are not restricted to powers of 2.
With this subroutine the only restrictions are:-

1. The (complex) series length N must have all factors less than or equal to 23,

2. The product of square-free factors of N must be less than or equal to 210.

As the list of possible piece lengths shows (Appendix) this puts very little restriction on the user.

Subroutine written originally by R.C.Singleton at Stanford Research Institute.
Converted for IBM 1800 by C.Clayson, W.J.Gould, M.Winbush.
Transferred to Honeywell 66 by W.K.Strudwick.

Input Parameters

A	Array containing real components of the data series.
B	Array containing imaginary components of the data series.
NTOT	Total number of complex variables.
N	Dimension of the current variable (for multivariate case).
NSPAN/N	Spacing of consecutive data values.
ISN	The sign of ISN is the sign of the complex exponential. The magnitude of ISN is 1.

Output Parameters

A — Array containing real components of resulting Fourier coefficients.
Multiply by 1/N to give usual scaling of coefficients.

B — Array containing imaginary components of resulting Fourier coefficients.
Multiply by 1/N to give usual scaling of coefficients.

Calling Subroutine

Single Variate Transform

For a single variate transform
NTOT = N = NSPAN = (Number of complex data values)

```
eg.  CALL FFT (A,B,N,N,N,1)
```

For real data, when array B would otherwise be filled with zeros, one can store the real data values alternately in arrays A and B, and then call subroutine RTRAN after FFT.
This allows N data values to be processed using a transform of dimension N/2.

```
eg.  NN=N/2
     CALL FFT (A,B,NN,NN,NN,1)
     CALL RTRAN (A,B,NN,1)
```

After scaling by 0.5/N the results in A and B are the Fourier cosine and sine coefficients.

i.e. $a_k = A(k+1)$ $k=0,1,2....NN$

$b_k = B(k+1)$ $k=0,1,2....NN$

The arrays A and B must be dimensioned in the main program with NN+1 and not NN as there are NN+1 coefficients to be calculated.

Multivariate Transform

For a multivariate transform the data is indexed according to the Fortran array element successor function. The subroutine is called once for each variate.
eg. a trivariate transform of A(N1,N2,N3), B(N1,N2,N3) is computed by

```
CALL FFT (A,B,N1*N2*N3,N1,N1,1)
CALL FFT (A,B,N1*N2*N3,N2,N1*N2,1)
CALL FFT (A,B,N1*N2*N3,N3,N1*N2*N3,1)
```

SUBROUTINE RTRAN(A,B,NN,ISN)

Function

Used with subroutine FFT to complete the Fourier Transform of 2*NN real data values and calculate the cosine and sine coefficients.
The original real data values before calling FFT were stored alternately in arrays A and B.

Subroutine written originally by R.C.Singleton at Stanford Research Institute.
Converted for IBM 1800 by W.J.Gould.
Transferred to Honeywell 66 by W.K.Strudwick.

Input Parameters

A,B — Before calling FFT these arrays contain real data values stored alternately.
eg. A(1), B(1), A(2), B(2),....A(NN), B(NN).
N.B. In main program arrays must be dimensioned (NN+1)

NN — There are 2*NN = N real data values.

ISN — If ISN = 1 completes Fourier Transform.
If ISN = -1 performs reverse transform.

Output Parameters

A — Array containing NN+1 cosine coefficients.
Multiply by 0.5/N to give usual scaling of coefficients.

B — Array containing NN+1 sine coefficients.
Multiply by 0.5/N to give usual scaling of coefficients.

Calling Subroutine

Single Variate Transform

For a single variate transform
For real data when array B would otherwise be filled with zeros, one can store the real data values alternately in arrays A and B, and then call subroutine RTRAN after FFT.
This allows N data values to be processed using a transform of dimension N/2.

```
eg.  NN=N/2
     CALL FFT (A,B,NN,NN,NN,1)
     CALL RTRAN (A,B,NN,1)
```

After scaling by 0.5/N the results in A and B are the Fourier cosine and sine coefficients.

i.e. a_k = A(k+1) k=0,1,2....NN

b_k = B(k+1) k=0,1,2....NN

The arrays A and B must be dimensioned in the main program with NN+1 and not NN as there are NN+1 coefficients to be calculated.

EXAMPLE OF SINGLE VARIATE TRANSFORM WITH COMPLEX DATA

```
010*#FRN *=(ULIB)LIBRARY/NIO,R
050C
060C SINGLE VARIATE FAST FOURIER TRANSFORM OF A COMPLEX DATA SERIES
070C
080       DIMENSION A(6),B(6)
090       DATA A/2.0,1.0,3.0,1.0,4.0,0.0/
100       DATA B/1.0,4.0,3.0,0.0,2.0,3.0/
110C
120       N=6
130       WRITE(6,3) N,(A(K),B(K),K=1,N)
140 3     FORMAT(' LENGTH OF COMPLEX DATA SERIES=',I6//,
150      &'   REAL COMPONENT   IMAGINARY COMPONENT'//(5X,F9.5,10X,F9.5))
160C
170       CALL FFT(A,B,N,N,N,1)
180C
190C SCALING
200C
210       DO 10 I=1,N
220       A(I)=A(I)/N
230 10    B(I)=B(I)/N
240C
250       WRITE(6,4) (A(K),B(K),K=1,N)
260 4     FORMAT(//' RESULTING FOURIER COEFFICIENTS'//,
270      &'   REAL COMPONENT   IMAGINARY COMPONENT'//(5X,F9.5,10X,F9.5))
280       STOP
290       END
```

```
*FRN
LENGTH OF COMPLEX DATA SERIES=      6

  REAL COMPONENT    IMAGINARY COMPONENT

       2.00000            1.00000
       1.00000            4.00000
       3.00000            3.00000
       1.00000            0.
       4.00000            2.00000
       0.                 3.00000


RESULTING FOURIER COEFFICIENTS

  REAL COMPONENT    IMAGINARY COMPONENT

       1.83333            2.16667
      -0.62201            0.33333
      -0.16667           -0.54466
       1.16667           -0.16667
      -0.16667           -1.12201
      -0.04466            0.33333
```

EXAMPLE OF SINGLE VARIATE TRANSFORM WITH REAL DATA CALLING FFT ONLY

```
010*#FRN *=(ULIB)LIBRARY/NIO,R
050C
060C SINGLE VARIATE FAST FOURIER TRANSFORM
070C OF A REAL DATA SERIES.
080C
090       DIMENSION A(6),B(6)
100       DATA A/2.0,1.0,3.0,1.0,4.0,0.0/
110       DATA B/6*0.0/
120C
130       N=6
140       WRITE(6,3) N,(A(K),B(K),K=1,N)
150 3     FORMAT(' LENGTH OF REAL DATA SERIES=',I6//,
160      &'   REAL COMPONENT   IMAGINARY COMPONENT'//(5X,F9.5,10X,F9.5))
170C
180       CALL FFT(A,B,N,N,N,1)
190C
200C SCALING
210C
220       DO 10 I=1,N
230       A(I)=A(I)/N
240 10    B(I)=B(I)/N
250C
260       WRITE(6,4) (A(K),B(K),K=1,N)
270 4     FORMAT(//' RESULTING FOURIER COEFFICIENTS'//,
280      &'   REAL COMPONENT   IMAGINARY COMPONENT'//(5X,F9.5,10X,F9.5))
290       STOP
300       END

*FRN
LENGTH OF REAL DATA SERIES=      6

  REAL COMPONENT    IMAGINARY COMPONENT

       2.00000            0.
       1.00000            0.
       3.00000            0.
       1.00000            0.
       4.00000            0.
       0.                 0.


RESULTING FOURIER COEFFICIENTS

  REAL COMPONENT    IMAGINARY COMPONENT

       1.83333            0.
      -0.33333            0.
      -0.16667            0.28868
       1.16667            0.
      -0.16667           -0.28868
      -0.33333            0.

DP/US/NIO/1                REL 1.0                      RTRAN/1
```

EXAMPLE OF SINGLE VARIATE TRANSFORM WITH REAL DATA CALLING FFT AND RTRAN

```
010*#FRN *=(ULIB)LIBRARY/NIO,R
050C
060C SINGLE VARIATE FAST FOURIER TRANSFORM OF REAL DATA SERIES.
070C
080C
090       DIMENSION A(4),B(4)
100       DATA A/2.0,3.0,4.0,0.0/
110       DATA B/1.0,1.0,0.0,0.0/
120C
130       N=6
140       NN=N/2
150       WRITE(6,3) N,(A(K),B(K),K=1,NN)
160 3     FORMAT(' LENGTH OF REAL DATA SERIES=',I6//,
170      &'    REAL DATA VALUES='//(5X,F9.5))
180C
190       CALL FFT(A,B,NN,NN,NN,1)
200       CALL RTRAN(A,B,NN,1)
210C
220C SCALING
230C
240       DO 10 I=1,NN+1
250       A(I)=A(I)*0.5/N
260 10    B(I)=B(I)*0.5/N
270C
280       WRITE(6,4) (A(K),B(K),K=1,NN+1)
290 4     FORMAT(//' RESULTING COSINE AND SINE COEFFICIENTS'//,
300      &'    COSINE                SINE          '//(5X,F9.5,10X,F9.5))
310       STOP
320       END

*FRN
LENGTH OF REAL DATA SERIES=      6

  REAL DATA VALUES=

      2.00000
      1.00000
      3.00000
      1.00000
      4.00000
      0.


RESULTING COSINE AND SINE COEFFICIENTS

   COSINE                SINE

     1.83333             0.
    -0.33333             0.00000
    -0.16667             0.28868
     1.16667             0.

DP/US/NIO/1               REL 1.0                    RTRAN/1
```

DATA SERIES LENGTHS FOR USE WITH FET

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
27 28 30 32 33 34 35 36 38 39 40 42 44 45 46 48 49 50 51 52 54 55 56 57 60
63 64 65 66 68 69 70 72 75 76 77 78 80 81 84 85 88 90 91 92 95 96 98 99

100 102 104 105 108 110 112 114 115 117 119 120 121 125 126 128 130 132
133 135 136 138 140 143 144 147 150 152 153 154 156 160 161 162 165 168
169 170 171 175 176 180 182 184 187 189 190 192 195 196 198 200 204 207
208 209 210 216 220 224 225 228 234 240 242 243 245 250 252 256 260 264
270 272 275 276 280 288 289 294 297 300 304 306 308 312 315 320 324 325
335 338 340 342 343 350 351 352 360 361 363 364 368 375 378 380 384 392
396 400 405 408 414 416 420 425 432 440 441 448 450 456 459 468 475 480
484 486 490 495 500 504 507 512 513 520 525 528 529 539 540 544 550 552
560 567 575 576 578 585 588 594 600 605 608 612 616 621 624 625 630 637
640 648 650 672 675 676 680 684 686 700 702 704 720 722 726 728 729 735
736 750 756 760 768 784 792 800 810 816 825 828 832 833 840 845 847 850
864 867 875 882 891 896 900 918 931 936 945 950 960 968 972 975 980

1000 1008 1014 1024 1026 1029 1050 1053 1058 1078 1080 1083 1088 1089 1100
1125 1127 1134 1152 1156 1176 1183 1188 1200 1210 1215 1216 1224 1225 1242
1250 1260 1274 1280 1296 1300 1323 1331 1344 1350 1352 1368 1372 1375 1377
1400 1404 1408 1440 1444 1445 1452 1458 1470 1472 1485 1500 1512 1521 1536
1539 1568 1573 1575 1584 1587 1600 1620 1625 1656 1664 1690 1694 1700 1701
1715 1728 1734 1750 1755 1764 1782 1792 1800 1805 1815 1836 1859 1863 1872
1875 1890 1900 1920 1936 1944 1960 2000 2016 2023 2025 2028 2048 2052 2057
2058 2100 2106 2116 2125 2156 2160 2166 2176 2178 2187 2197 2200 2205 2250
2268 2299 2304 2312 2352 2366 2375 2376 2400 2401 2420 2430 2432 2448 2450
2475 2484 2500 2520 2527 2535 2548 2560 2592 2600 2601 2625 2645 2648 2662
2673 2688 2700 2704 2744 2750 2754 2800 2808 2816 2835 2875 2880 2888 2890
2904 2916 2925 2940 2944 3000 3024 3025 3042 3072 3078 3087 3125 3126 3150
3159 3174 3179 3200 3240 3249 3250 3267 3328 3375 3380 3388 3400 3402 3430
3456 3468 3500 3528 3564 3584 3600 3610 3645 3672 3675 3703 3726 3750 3773
3780 3800 3840 3872 3888 3920 3969 3971 3993 4000 4032 4050 4056 4096 4104
4116 4125 4131 4200 4212 4225 4232 4250 4312 4320 4332 4352 4356 4374 4375
4394 4410 4455 4459 4500 4536 4563 4608 4617 4624 4704 4725 4732 4750 4752
4761 4800 4802 4840 4860 4864 4875 4900 4913 4968 5000 5096 5103 5120 5145
5184 5202 5250 5265 5292 5324 5346 5376 5400 5408 5445 5488 5500 5508 5589
5616 5625 5632 5670 5760 5776 5780 5808 5831 5832 5880 5888 5929 6000 6048
6050 6075 6084 6125 6144 6156 6174 6250 6272 6300 6318 6348 6400 6480 6498
6500 6517 6534 6561 6591 6615 6655 6656 6750 6760 6776 6804 6859 6860 6875
6912 6936 7000 7056 7128 7168 7200 7203 7220 7225 7290 7344 7350 7425 7452
7500 7546 7560 7605 7680 7744 7776 7803 7875 7889 7938 7986 8000 8019 8064
8100 8112 8125 8192 8232 8262 8281 8424 8448 8450 8464 8500 8505 8575 8640
8664 8704 8712 8748 8750 8775 8788 8820 8918 8960 9000 9025 9072 9075 9126
9216 9234 9248 9261 9317 9375 9408 9450 9464 9477 9500 9522 9600 9604 9720
9728 9747 9800 9801 9826 9984

10000 10125 10206 10240 10290 10368 10404 10500 10584 10625 10648 10692
10752 10800 10816 10935 10976 10985 11000 11016 11025 11178 11250 11264
11340 11520 11552 11560 11664 11776 11858 11875 11907 11979 12000 12005
12096 12150 12167 12168 12250 12288 12312 12348 12375 12393 12500 12544
12600 12636 12675 12696 12800 12960 13000 13056 13068 13122 13125 13182
13225 13230 13310 13312 13365 13500 13608 13689 13718 13720 13750 13824
13851 13872 14000 14112 14161 14175 14256 14283 14336 14375 14400 14406
14440 14450 14580 14625 14641 14700 14739 14904 15000 15092 15125 15309
15360 15379 15435 15488 15552 15806 15625 15750 15795 15876 15972 16000
16038 16128 16200 16250 16335 16384 16464 16524 16562 16767 16807 16848
16875 16928 17000 17010 17150 17280 17303 17408 17424 17496 17500 17576

17640 17689 17836 18000 18050 18144 18225 18252 18375 18432 18468 18496
18522 18634 18750 18816 18900 18954 19000 19200 19208 19440 19456 19494
19600 19602 19652 19663 19773 19845 19965 20000 20250 20412 20449 20480
20577 20580 20625 20736 20808 21000 21125 21168 21250 21296 21384 21504
21600 21609 21632 21870 21875 21952 21970 22032 22050 22275 22356 22500
22528 22627 22680 22815 23040 23104 23328 23409 23552 23625 23750 23814
23958 24000 24010 24037 24167 24192 24200 24300 24334 24336 24375 24500
24565 24576 24696 24736 25000 25080 25272 25289 25344 25515 25600 25725
25920 25921 25992 26136 26244 26250 26325 26364 26411 26460 26620 26624